Essential Model Documentation (EMD)

Authors	David Hassell, Karl Taylor, Guillaume Levavasseur, Charlotte Pascoe, Alex Crawford, Paul Durack, Daniel Ellis, Christopher Goddard, Andras Horanyi, Bryan Lawrence, Zebedee Nicholls, Martina Stockhause, Laurent Troussellier
Contributors	Sasha Ames, Katharina Berger, Diana Gergel, Maria Molina, Atef Ben Nasser, James Pope, Martin Schupfner, Lina Teckentrup, Jeremy Walton
Version	1.0
Date	2025-12-03

Table of contents

List of figures

1. Introduction

The Essential Model Documentation (EMD) is a high-level description of an earth system model.

It is intended to contain information about model configuration that may be helpful to the communities who expect to make use of the model output, whilst not imposing burdensome requirements on those providing the data.

It is not intended to contain all information about a model. More detailed model documentation than that provided by the EMD should be found in the references cited as part of the EMD, or from other external sources.

The EMD is collected for the model as a whole in section 2 (Top-level model), and for each of the model's components in section 3 (Model components).

This document is intended to be a human-readable specification with initial, but not complete, controlled vocabularies. It does not place any constraint on the tools and file-serialisation that will be required to create, store, and access EMD content. In particular, it is expected that the controlled vocabularies will evolve to meet the needs of the models being described.

1.1. *Application to CMIP7*

The EMD has been designed to be applicable to any earth system model, nonetheless some guidance has been included in this version for its use within the CMIP7 project (Dunne et al., 2025), for which EMD was originally developed. This is indicated in the specification as "***Note for CMIP7***", and these particular notes and instructions may be ignored when using EMD within other projects.

The EMD will be considered especially valuable if it is provided by all models. To ensure this, it has been agreed that the EMD will be a mandatory requirement for CMIP7 participation, and the registration of a CMIP7 model will not be possible unless its EMD has been provided.

An on-line creation tool will be used for CMIP7 model registration, and this tool will collect the content that will be recorded in the EMD. The EMD will be reviewed, and when it is accepted the model registration can be completed. The on-line tool will also enable those documenting a model to import documentation from earlier registered models, model components, and grids, which can then be edited, if required.

2. Top-level model

A description of the top-level model as whole.

A top-level model is described by the following properties, for which *underlined* values in the examples are taken from section 7 (Controlled vocabularies):

- **name**
  - The name of the top-level model.
  - The name may include an indication of the top-level model's family (given by the model **family** property), and should also specify a particular version of the top-level model.
  - ***Note for CMIP7****: This name will be registered as the model's source_id*.
  - E.g. *HadGEM3-GC31-HH*
- **family**
  - The top-level model's "family" name. A family of models share much of their code bases, but may be configured in different ways (e.g. different resolutions, parameter choices, or the inclusion or not of particular model components). See, for instance, Masson and Knutti (2011) for an example of how the family can be used to inform model genealogies.
  - A family name often persists across model versions and may be the first part of its full name, as given by the top-level model's **name** property.
  - Omit if there is no such family for the top-level model.
  - E.g. *HadCM2*
  - E.g. *HadGEM3*
  - E.g. *CCSM*
- **dynamic_components**
  - The model components that are dynamically simulated within the top-level model.
  - Taken from a standardised list: 7.1 (component CV).
  - These components must be further described by the model component properties, as discussed in section 3 (Model components).
  - There must be at least one dynamic component.
  - E.g. *atmosphere*, *land_ice*, *land_surface*
- **prescribed_components**
  - The components that are represented in the top-level model with prescribed values.
  - Taken from a standardised list: 7.1 (component CV).
  - These components are typically represented by constant or time varying values or fields that are provided as input to the top-level model.
  - Omit if there are no prescribed components.
  - E.g. *atmospheric_chemistry*, *ocean*, *sea_ice*
- **omitted_components**
  - The components that are wholly omitted from the top-level model.
  - Taken from a standardised list: 7.1 (component CV).
  - Omit if there are no omitted components.
  - E.g. *aerosol*, *ocean_biogeochemistry*
- **description**
  - A scientific overview of the top-level model.
  - The description should include a brief mention of all the components listed in the 7.1 (component CV), whether dynamically simulated, prescribed, or omitted.
  - Dynamically simulated components, listed by the **dynamic_components** property, are described elsewhere, see section 3 (Model components), so only a very short, high-level overview is required in this description. For instance "*The model includes a global process-based model of the land surface and the terrestrial biosphere that calculates water, energy, and carbon fluxes between the surface and the atmosphere*", and this description should also include the component's name, if appropriate.

- Prescribed components, listed by the **prescribed_components** property, are not described elsewhere, so information on how they are treated should be included in the description. For instance "*The model requires that a monthly mean, zonally averaged ozone field be prescribed because the model does not include an interactive atmospheric chemistry component*".
- Components that are wholly omitted, listed by the **omitted_components** property, must be noted as such. For instance, "*The model has no treatment of atmospheric chemistry*".

– **calendar**
  - The calendar, or calendars, that define which dates are permitted in the top-level model.
  - Taken from a standardised list: 7.2 (calendar CV).
  - ***Note for CMIP7****: Only the set of calendars that are adopted in CMIP7 simulations should be provided, identifying which calendar is most commonly used by listing it first. For instance, if standard applies for historical and future simulations and 365_day for paleo-experiments, then calendar would be set to standard, 365_day*.
  - E.g. *360_day*
  - E.g. *standard, 365_day*

– **release_year**
  - The year in which the top-level model being documented was released or first used for published simulations.
  - E.g. *2016*

– **references**
  - One or more references to published work describing the top-level model as a whole.
  - Each reference is described by the properties listed in section 5 (References).

3. Model components

A description of an individual model component of the top-level model.

Eight model components are defined that somewhat independently account for different sets of interactive processes: aerosol, atmosphere, atmospheric chemistry, land surface, land ice, ocean, ocean biogeochemistry, and sea ice. The interactive processes covered by each component are described in section 7.1 (component CV).

Each component is characterised in section 2 (Top-level model) as "dynamic", "prescribed", or "omitted"; but only model components that dynamically simulate their processes are described in this section of the EMD. Relationships between dynamically simulated model components are indicated by specifying that they are "embedded in" or "coupled with" other components, as described in see section 3.1 (Relationships between model components).

Note for CMIP7*: The component types in the 7.1 (component CV) have similar names and definitions to the CMIP "realms" (Durack et al., 2025), but the context in which they are used is different. An EMD component type defines a set of physical processes that are simulated by one model component; whereas as one or more CMIP realms are assigned to an individual output variable according to which sets of processes the variable is physically related to, rather than which model component created it. The CMIP realms for an output variable often include the EMD component type that created it, but this is not always the case.*

A model component is described by the following properties, for which *underlined* values in the examples are taken from section 7 (Controlled vocabularies):

- **component**
  - The type of the model component.
  - Taken from a standardised list: 7.1 (component CV).
  - E.g. *aerosol*
- **name**
  - The name of the model component.
  - If the component is embedded in a host component and has no commonly recognised name, then a name can be constructed by combining the host component's **name** with this component's **component** type, separated by a hyphen.
  - E.g. *BISICLES-UKESM-ISMIP6*
  - E.g. *MOSES2*
  - E.g. *HadAM3-aerosol*
- **family**
  - The model component's "family" name. For a component, its family members should all share much of their code bases, but the members may be configured in different ways (e.g. different resolutions, parameter choices, or the inclusion or not of particular physical process). See, for instance, Masson and Knutti (2011) for an example of how the family can be used to inform model genealogies.
  - Omit if there is no such family for the model component.
  - E.g. *BISICLES*
  - E.g. *CLM*
- **description**
  - A scientific overview of the model component.
  - The description should summarise the key processes simulated by the model component.
- **references**
  - One or more references to published work for the model component.
  - Each reference is described by the properties listed section 5 (References).
- **code_base**
  - The location of the source code for the model component.

    - The location must be a fully qualified uniform resource locator (URL), and could be a unique persistent identifier (DOI).
    - If the code base is in a version control repository (such as a git repository) then the URL should identify a specific point in the repository's history.
    - Omit if the source code is not available.
- **embedded_in**
    - The host model component for which this component is "embedded in".
    - The host component is identified by its **component** property from the standardised list: 7.1 (component CV).
    - See section 3.1 (Relationships between model components) for the definition of embedded and host components.
    - Note that a component which is embedded in another component can not also be coupled with any components.
    - Omit when this component is not embedded in another component.
    - E.g. for an atmospheric chemistry model component that is embedded as part of the atmosphere component: *atmosphere*
- **coupled_with**
    - The model components for which this component is "coupled with".
    - Each coupled component is identified by its **component** property from the standardised list: 7.1 (component CV).
    - See section 3.1 (Relationships between model components) for the definition of coupled components.
    - Note that a component which is embedded in another component can not also be coupled with any components.
    - Omit when this component is not coupled with any other components.
    - E.g. In some cases for a ocean component: *atmosphere*, *land_ice*
- **horizontal_computational_grid**
    - A description of the model component's horizontal computational grid, i.e. the grid on which the component is integrated.
    - The grid is described by defining the properties listed in section 4.1.1 (Horizontal computational grid).
    - Omit if the model component has no horizontal computational grid.
- **vertical_computational_grid**
    - A description of the model component's computational vertical grid, i.e. the grid on which the component is integrated.
    - The grid is described by defining the properties listed in section 4.2.1 (Vertical computational grid).
    - Omit if the model component has no vertical computational grid.

3.1. *Relationships between model components*

All dynamically simulated components within a top-level model interact with each other either directly or indirectly. The EMD only identifies which components directly interact, characterising each component as either being "embedded in" or "coupled with" other components, using the **embedded_in** and **coupled_with** model component properties respectively. See section 3 (Model components).

***Note**: Beyond the characterisation of "embedded in" or "coupled with", the EMD does not include further information about the nature of component interactions (e.g. it does not state which physical quantities are exchanged between components, nor how frequently these exchanges occur).*

The following characteristics can be used to help determine whether a component is "embedded in" or "coupled with" another component:

An embedded model component typically:

- shares the same horizontal grid as the host component;
- is constructed such that the code representing the component would be exceedingly difficult to extract and transfer to another top-level model;
- is coded such that within a single time-step, interactions between it and its host might involve exchanges of information affecting both;
- cannot be easily "driven" independently of its host (e.g. it cannot be run by itself in stand-alone "offline" mode with prescribed external conditions imposed);
- may not have a recognisable name or version number (other than those of the host component).

A coupled model component typically:

- is identified using a recognisable name, and often a version number (e.g. "MOM5", the fifth version of the Modular Ocean Model);
- relies on a variable calculated by another component or calculates a variable relied on by another component, or interacts at regular intervals with other components by exchanging quantities needed by each to advance a simulation (such as fluxes of mass, momentum, energy, etc.). The shared quantities may either pass through a coupler (i.e. code specially adapted to represent the exchanges between the various components) or be conveyed by other means;
- is coded such that it is relatively isolated from other parts of a top-level model code and (by design) might be extracted to be adopted in another top-level model.

The distinction between these two types may not always be obvious, and in some cases a somewhat arbitrary choice may have to be made. This could be the case, for instance, when a component satisfies characteristics from both interaction types (e.g. it is possible for a component with a recognisable name and version number to be embedded in a host, rather than coupled).

There is great flexibility in the ways in which model components can potentially be coupled and embedded, and how this actually is done is an important part of the top-level model formulation.

An embedded component has a single host component, which is recorded in the embedded component's **embedded_in** property. Multiple components can be embedded in the same host, but a host does not record those components that are embedded in it. The host of an embedded component may itself be embedded in another host component, forming a hierarchy of embedding that has a single "superior" host at the top, which is not embedded in any other component. The simplest hierarchy contains a single component, which is the hierarchy's superior host, even though it contains no embedded components. For instance, in **figure 1** there are four hierarchies, with superior host components of atmosphere, ocean, land surface, and land ice; the last two of which represent hierarchies containing a single component.

A superior host component must be coupled with at least one other superior host component, unless there are no others. Coupling, unlike embedding, is symmetrical in that if component A is coupled with component B, then component B must also be coupled with component A; and this would be recorded in the **coupled_with** properties of both components. An embedded component can not be coupled with any other components. For instance, in **figure 1**, coupling can only occur between the four superior host components of atmosphere, ocean, land surface, and land ice.

Sometimes it may not be obvious whether component A is embedded in component B or vice versa, since they are essentially coded together. For example, when the aerosol and atmospheric chemistry components are integrated together, which one should be considered to be embedded in the other? The model developers are free to decide which of a pair of components that are integrated together is considered to be embedded in the other, and the restriction that only superior host components can be coupled with each other may help to decide how host components are chosen.

***Note**: An embedding hierarchy is not allowed to have cyclical embedding, meaning that a host component can not itself be embedded in a component that is lower in the same hierarchy (i.e. further away from the superior host). For instance, in **figure 1**, the atmosphere component would not be allowed to be embedded in either the atmospheric chemistry or aerosol component.*

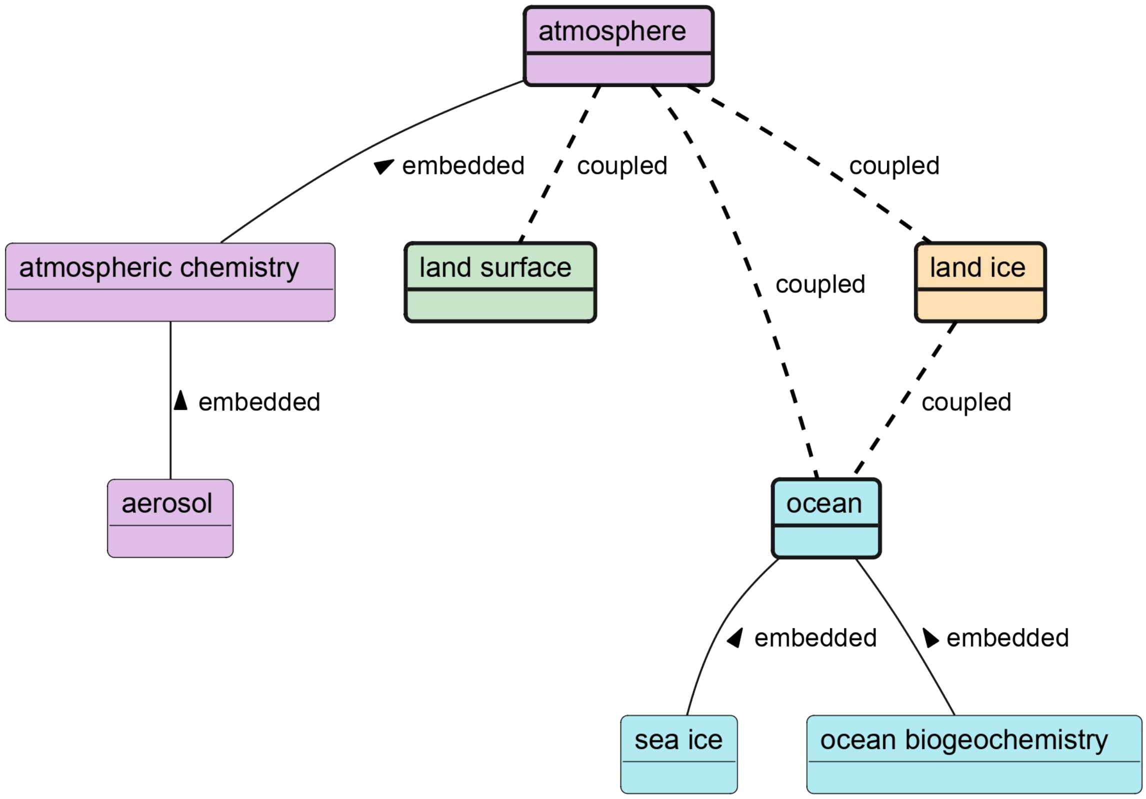

Figure 1 *The relationships between model components for a hypothetical top-level model. Coupling occurs between the four superior host components (atmosphere, ocean, land surface, and land ice), but not in every possible combination (the land surface is only coupled with the atmosphere). The other four components are each embedded in a host component, and are therefore not coupled with any other components. The atmospheric chemistry, sea ice, and ocean biogeochemistry components are each embedded directly in the superior host component of their hierarchy (atmosphere, ocean, and ocean respectively); but the aerosol component is embedded in atmospheric chemistry (not a superior host component), which in turn is embedded in the hierarchy's atmosphere superior host component.*

4. Model component grids

Properties that provide a description of the grids of a model component defined in section 3 (Model components).

Documentation of the computational grid of each model component, i.e. the grid on which the component is integrated, is split into separate horizontal and vertical parts. Each part is described with a standardised specification that is based on selections from controlled vocabularies, or the provision of numerical values or text descriptions. See sections 4.1 (Horizontal grid) and 4.2 (Vertical grid).

Note*: A model component may be mapped to an "output" grid, that is different from the computational grid, in order to facilitate subsequent analysis. For example, data computed on a terrain-following vertical computational grid might be mapped to constant pressure levels, and data on a variable resolution horizontal computational grid might be mapped to a regular 1 degree grid. These output grids are not, however, included in the EMD description; the EMD only describes the computational grids on which the model components are integrated. However, the horizontal grid cells description of section 4.1.3 (Horizontal grid cells) may also be used independently of the rest of the EMD to describe an output grid.*

4.1. Horizontal grid

4.1.1. Horizontal computational grid

A model component's horizontal computational grid is composed of one or more horizontal subgrids, on which different sets of variables are calculated. When the computational grid relies on more than one subgrid, it is referred to as a "staggered" grid. For most staggered grids, the velocity-related variables are calculated on a subgrid offset from the mass-related variables (e.g. pressure, temperature, water vapour and other mass constituents). See section 3 (Model components).

Examples of horizontal computational grid descriptions are given in section 6 (Examples).

A horizontal computational grid is described by the following properties, for which *underlined* values in the examples are taken from section 7 (Controlled vocabularies):

- **arrangement**
  - A characterisation of the grid staggering defining the relative positions of computed mass-related and velocity-related variables.
  - Taken from a standardised list: 7.3 (arrangement CV).
  - E.g. *arakawa_c*
- **horizontal_subgrids**
  - All of the subgrids, of which there must be at least one, used to construct the horizontal computational grid.
  - Each subgrid is described by the properties listed in section 4.1.2 (Horizontal subgrid), which includes the association of the subgrid with one or more variable types (mass-related, velocity-related, etc.), consistent with the **arrangement** property.

4.1.2. Horizontal subgrid

A horizontal subgrid describes the grid cells at one of the stagger positions of a horizontal computational grid. Often the locations of mass-related and velocity-related variables differ, so more than one horizontal subgrid will be defined as part of a horizontal computational grid. See section 4.1.1 (Horizontal computational grid).

A horizontal subgrid is described by the following properties, for which *underlined* values in the examples are taken from section 7 (Controlled vocabularies):

- **cell_variable_type**
  - The types of physical variables that are carried at, or representative of conditions at, the cells described by this horizontal subgrid.
  - Taken from a standardised list: 7.4 (cell_variable_type CV).
  - E.g. *mass*
  - E.g. *x_velocity*
  - E.g. *mass*, *x_velocity*, *y_velocity*
  - E.g. *mass*, *velocity*
- **horizontal_grid_cells**
  - A description of the characteristics and location of the grid cells of this subgrid.
  - The horizontal grid cells are described by the properties listed in section 4.1.3 (Horizontal grid cells).

4.1.3. *Horizontal grid cells*

Horizontal grid cells are described by a coordinate system, cell resolutions, as well as a number of other grid features. The description does not include any information on whether or not the grid cells form part of a model component's computational grid, and so may be used to describe an arbitrary output grid. If there is a relationship with a computational grid, then may be provided by a parent horizontal subgrid. See sections 4.1.1 (Horizontal computational grid) and 4.1.2 (Horizontal subgrid).

Horizontal grid cells are described by the following properties, for which *underlined* values in the examples are taken from section 7 (Controlled vocabularies):

- **region**
  - The geographical region, or regions, over which the component is simulated.
  - A region is a contiguous part of the Earth's surface, and may include areas for which no calculations are made (such as ocean areas for a land surface component).
  - Taken from a standardised list: 7.5 (region CV).
  - E.g. *global*
  - E.g. *antarctica*, *greenland*
  - E.g. *limited_area*
- **grid_type**
  - The horizontal grid type, i.e. the method of distributing grid cells over the region.
  - Taken from a standardised list: 7.6 (grid_type CV).
  - E.g. *regular_latitude_longitude*
  - E.g. *tripolar*
- **description**
  - A description of the grid.
  - A description is only required if there is information that is not covered by any of the other properties.
  - Omit when not required.
- **grid_mapping**
  - The name of the coordinate reference system of the horizontal coordinates.
  - Taken from a standardised list: 7.7 (grid_mapping CV).
  - E.g. *latitude_longitude*
  - E.g. *lambert_conformal_conic*
- **temporal_refinement**
  - The grid temporal refinement, indicating how the distribution of grid cells varies with time.
  - Taken from a standardised list: 7.8 (temporal_refinement CV).
  - E.g. *static*
- **x_resolution**
  - The size of grid cells in the X direction.
  - Cells for which no calculations are made are included (such as ocean areas for a land surface component).

  - The X direction for a grid defined by spherical polar coordinates is longitude.
  - The value’s physical units are given by the **horizontal_units** property.
  - Report only when cell sizes are identical or else reasonably uniform (in their given units). When cells sizes in the X direction are not identical but vary by less than 25% of the largest size, a representative value should be provided and this fact noted in the **description** property.
  - Omit when the smallest cell size is less than 75% of the largest cell size, in the X direction.
  - E.g. *3.75*
- **y_resolution**
  - The size of grid cells in the Y direction.
  - Cells for which no calculations are made are included (such as ocean areas for a land surface component).
  - The Y direction for a grid defined by spherical polar coordinates is latitude.
  - The value’s physical units are given by the **horizontal_units** property.
  - Report only when cell sizes are identical or else reasonably uniform (in their given units). When cells sizes in the Y direction are not identical but vary by less than 25% of the largest size, a representative value should be provided and this fact noted in the **description** property.
  - Omit when the smallest cell size is less than 75% of the largest cell size, in the Y direction.
  - E.g. *2.5*
- **horizontal_units**
  - The physical units of the **x_resolution** and **y_resolution** property values.
  - Taken from a standardised list: 7.9 (horizontal_units CV).
  - Omit when neither of the **x_resolution** and **y_resolution** properties are set.
  - E.g. *km*
  - E.g. *degree*
- **southernmost_latitude**
  - The southernmost latitude coordinate of the grid cells, in degrees north.
  - The value must be greater than or equal to –90 and less than or equal to 90.
  - Cells for which no calculations are made are included (such as ocean areas for a land surface component).
  - The southernmost coordinate latitude may be shared by multiple cells.
  - Omit when not applicable, as could be the case for grids with an adaptive temporal refinement, or for grids that do not map to a sphere.
  - E.g. *–89.5*
- **westernmost_longitude**
  - The westernmost longitude coordinate, in degrees east, of the southernmost grid cell or cells.
  - The value must be greater than or equal to 0 and strictly less than 360.
  - Cells for which no calculations are made are included (such as ocean areas for a land surface component).
  - The westernmost longitude coordinate is the smallest longitude (in the range 0 to 360) of the cells that share the coordinate latitude given by the **southernmost_latitude** property.
  - Omit when not applicable, as could be the case for grids with an adaptive temporal refinement, or for grids that do not map to a sphere.
  - E.g. *0.5*
- **n_cells**
  - The total number of cells in the horizontal grid.
  - Cells for which no calculations are made are included (such as ocean areas for a land surface component).
  - Omit when not constant (such as could be the case for grids with an adaptive temporal refinement).

- E.g. *265160*

- **truncation_method**
  - The method for truncating the spherical harmonic representation of a spectral model.
  - Taken from a standardised list: 7.10 (truncation_method CV).
  - Omit when not applicable.
  - E.g. *triangular*
- **truncation_number**
  - The zonal (east-west) wave number at which a spectral model is truncated.
  - Omit when not applicable.
  - E.g. *63*

4.2. *Vertical grid*

4.2.1. *Vertical computational grid*

A model component's vertical computational grid defines the layers in which mass-related variables (e.g. pressure, temperature, water vapour and other mass constituents) are calculated. See section 3 (Model components).

Note*: The vertical computational grid is not divided into subgrids, as is done in the horizontal case (see section 4.1.1 (Horizontal computational grid)), because there are no current alternatives for the placement of vertical velocity-related variables, other than at the mass layer boundaries.*

In general, a model component has a single vertical computational grid, for which most of the grid-defining properties can be unambiguously provided. However, some model components may have different and wholly independent vertical grids for different physical processes. For instance, this could occur for a land surface model with soil, lake and snow sub-models. In such cases it may not be possible to provide some of the properties that define the component's vertical computational grid, because a single value might not apply to the grids of all of the different physical processes. When this happens, the **description** property should be used instead to record the relevant information that applies to each physical process.

Examples of vertical computational grid descriptions are given in section 6 (Examples).

A vertical computational grid is described by the following properties, for which *underlined* values in the examples are taken from section 7 (Controlled vocabularies):

- **vertical_coordinate**
  - The coordinate type of the vertical grid.
  - Taken from a standardised list: 7.11 (vertical_coordinate CV).
  - Omit if different physical processes have grids with different vertical coordinate types, noting this in the **description** property.
  - E.g. *height*
- **description**
  - A description of the vertical grid.
  - A description is only required if there is information that is not covered by any of the other properties.
  - Omit when not required.
- **n_z**
  - The number of mass layers (i.e. grid cells) in the vertical grid.
  - Omit if different physical processes have grids with different numbers of layers, noting this in the **description** property.
  - Omit if the number of layers varies in time or across the horizontal grid, in which case the **n_z_range** property should be considered instead.
  - E.g. *70*
- **n_z_range**

    - The minimum and maximum number of mass layers in a vertical grid for which the number of layers varies in time or across the horizontal grid.
    - Omit if different physical processes have grids with different ranges, noting this in the **description** property.
    - Omit when the **n_z** property has been set.
    - Omit if there is otherwise not a well defined range, noting this in the **description** property.
    - E.g. *5, 15*
- **top_layer_thickness**
    - The thickness, in metres, of the top model mass layer, i.e. the layer furthest away from the centre of the Earth.
    - If the thickness varies in time or across the horizontal grid, then a nominal or representative maximum value should be provided (e.g. as could be the case for an atmosphere grid with air pressure-based vertical coordinates).
    - Omit if different physical processes have grids with different top layer thicknesses, noting this in the **description** property.
    - Omit if there is no nominal or representative maximum value, noting this in the **description** property.
    - E.g. *150*
- **bottom_layer_thickness**
    - The thickness, in metres, of the bottom model mass layer, i.e. the layer closest to the centre of the Earth.
    - If the thickness varies in time or across the horizontal grid, then a nominal or representative maximum value should be provided (e.g. as could be the case for an atmosphere grid with air pressure-based vertical coordinates).
    - Omit if different physical processes have grids with different bottom layer thicknesses, noting this in the **description** property.
    - Omit if there is no nominal or representative maximum value (e.g. as could be the case for a land ice model for which the bottom layer can expand to hold any amount of accumulated mass), noting this in the **description** property.
    - E.g. *10*
- **total_thickness**
    - The total thickness, in metres, spanned by all the model layers.
    - If the thickness varies in time or across the horizontal grid, then a nominal or representative maximum value should be provided (e.g. as could be the case for an atmosphere grid with air pressure-based vertical coordinates).
    - Omit if different physical processes have grids with different total thicknesses, noting this in the **description** property.
    - Omit if there is no nominal or representative maximum value (e.g. as could be the case for a land ice model for which can expand to hold any amount of accumulated mass), noting this in the **description** property.
    - E.g. for a global ocean grid, a nominal total thickness could be *5500*
    - E.g. for an air pressure-based atmosphere grid that goes up to 100 pascals, a nominal total thickness could be *48000*
    - E.g. for a land surface model, the total thickness could be *43*

5. References

A reference is the citation of a published work that describes some or all aspects of the top-level model or a model component to which it applies. The purpose of citing a reference is to provide a source of more detailed documentation than can be found in the overview that is contained in the EMD. See sections 2 (Top-level model) and 3 (Model components).

A reference is defined by the following properties:

- **citation**
  - A human-readable citation for the work.
  - The citation may use any bibliographic style.
  - E.g. *Smith, R. S., Mathiot, P., Siahaan, A., Lee, V., Cornford, S. L., Gregory, J. M., et al. (2021). Coupling the U.K. Earth System model to dynamic models of the Greenland and Antarctic ice sheets. Journal of Advances in Modeling Earth Systems, 13, e2021MS002520. https://doi.org/10.1029/2021MS002520*
- **doi**
  - A unique persistent identifier (DOI) for the work.
  - A DOI must be provided, even when it is also included as part of the **citation** property.
  - The DOI must be a uniform resource locator (URL) starting with “https://doi.org/”.
  - A reference that does not already have a DOI (as could be the case for some technical reports, for instance) must be given one, e.g. with a service like Zenodo.
  - E.g. *https://doi.org/10.1029/2021MS002520*

6. Examples

Here are a limited number of model component examples, and grid-only examples, that are based on some CMIP6 models.

In the examples, *underlined* values are taken from section 7 (Controlled vocabularies).

Note: *These examples are for illustrative purposes only, and must not be considered as definitive descriptions of these model components.*

6.1. *Land Surface component*

- **component**: *land_surface*
- **name**: CLM4
- **family**: CLM
- **description**: The model represents several aspects of the land surface including surface heterogeneity and consists of components or submodels related to land biogeophysics, the hydrological cycle, biogeochemistry, human dimensions, and ecosystem dynamics. Spatial land surface heterogeneity in CLM is represented as a nested subgrid hierarchy in which grid cells are composed of multiple landunits, snow/soil columns, and PFTs. Each grid cell can have a different number of landunits, each landunit can have a different number of columns, and each column can have multiple PFTs. Biogeophysical processes are simulated for each subgrid landunit, column, and PFT independently and each subgrid unit maintains its own prognostic variables. The same atmospheric forcing is used to force all subgrid units within a grid cell. The surface variables and fluxes required by the atmosphere are obtained by averaging the subgrid quantities weighted by their fractional areas.
- **references**
  - **citation**: Ke, Y., Leung, L. R., Huang, M., Coleman, A. M., Li, H., and Wigmosta, M. S.: Development of high resolution land surface parameters for the Community Land Model, Geosci. Model Dev., 5, 1341–1362, https://doi.org/10.5194/gmd-5-1341-2012, 2012.
  - **doi**: https://doi.org/10.5194/gmd-5-1341-2012
- **embedded_in**: *atmosphere*
- **horizontal_computational_grid**
  - **arrangement**: *arakawa_a*
  - **horizontal_subgrid**
    - **cell_variable_type**: *mass*
    - **horizontal_grid_cells**
      - **grid_type**: *regular_latitude_longitude*
      - **grid_mapping**: *latitude_longitude*
      - **region**: *global*
      - **temporal_refinement**: *static*
      - **x_resolution**: 1.25
      - **y_resolution**: 0.9
      - **horizontal_units**: *degree*
      - **n_cells**: 55296
      - **southernmost_latitude**: –89.65
      - **westernmost_longitude**: 0.0
- **vertical_computational_grid**
  - **description**: Vegetated, wetland, and glacier landunits have 15 vertical layers. Lakes have 10 layers. Snow can have up to 5 layers.
  - **vertical_coordinate**: *depth*

6.2. *Land Ice component*

- **component**: *land_ice*
- **name**: BISICLES-UKESM-ISMIP6-1.0
- **family**: BISICLES
- **description**: UniCiCles (Unified Model-CISM-BISICLES) is a package combining BISICLES with an interface that obtains boundary conditions from Unified Model or JULES data, using code derived from the Glint interface of the Glimmer-CISM ISM. BISICLES uses the adaptive-mesh Chombo libraries. All cells in the mesh are rectangles that may be recursively refined by subdivision into four smaller cells with the same aspect ratio. The configuration of BISICLES approximates the momentum equations using the “shelfy-stream” approximation with simplified vertical shear strains included in the effective viscosity, often referred to as SSA*. Basal traction is set to zero beneath floating ice and modelled using power laws beneath grounded ice. Greenland ice sheet uses a linear drag law everywhere while AIS uses a cubic law far upstream from the grounding line, which tends to a Coulomb friction law near the grounding line.
- **references**
    - **citation**: Smith, R. S., Mathiot, P., Siahaan, A., Lee, V., Cornford, S. L., Gregory, J. M., et al. (2021). Coupling the U.K. Earth System model to dynamic models of the Greenland and Antarctic ice sheets. Journal of Advances in Modeling Earth Systems, 13, e2021MS002520. https://doi.org/10.1029/2021MS002520
    - **doi**: https://doi.org/10.1029/2021MS002520
- **coupled_with**: *atmosphere*, *land_surface*, *ocean*
- **horizontal_computational_grid**
    - **arrangement**: *arakawa_c*
    - **horizontal_subgrid**
        - **cell_variable_type**: *mass*, *x_velocity*, *y_velocity*
        - **horizontal_grid_cells**
            - **description**: Greenland ice sheet (GrIS) is modelled with 9.6 km square base cells that may subdivide to 1.2 km, and Antarctic ice sheet (AIS) with 8 km that may subdivide to 2 km. The meshes are updated every 8 timesteps for GrIS and 4 for AIS, allowing the resolution to evolve with the ice dynamics.
            - **grid_type**: *plane_projection_grid*
            - **grid_mapping**: *polar_stereographic*
            - **region**: *greenland*, *antarctica*
            - **temporal_refinement**: *adaptive*
- **vertical_computational_grid**
    - **description**: The bottom level of the snowpack, which starts 6 metres below the snow surface, expands in thickness to hold as much mass as the column accumulates.
    - **vertical_coordinate**: *land_ice_sigma_coordinate*
    - **n_z**: 20
    - **top_level_thickness**: 0.04

6.3. *Grid: Regular Arakawa C grid/atmosphere_hybrid_height_coordinate*

- **horizontal_computational_grid**
    - **arrangement**: *arakawa_c*
    - **horizontal_subgrid**
        - **cell_variable_type**: *mass*
        - **horizontal_grid_cells**
            - **grid_type**: *regular_latitude_longitude*
            - **grid_mapping**: *latitude_longitude*

  - **region**: _global_
  - **temporal_refinement**: _static_
  - **x_resolution**: 1.0
  - **y_resolution**: 2.0
  - **horizontal_units**: _degree_
  - **southernmost_latitude**: –89.5
  - **westernmost_longitude**: 1.0
  - **n_ cells**: 32400
- **horizontal_subgrid**
  - **cell_variable_type**: _x_velocity_
  - **horizontal_grid_cells**
    - **grid_type**: _regular_latitude_longitude_
    - **grid_mapping**: _latitude_longitude_
    - **region**: _global_
    - **temporal_refinement**: _static_
    - **x_resolution**: 1.0
    - **y_resolution**: 2.0
    - **horizontal_units**: _degree_
    - **southernmost_latitude**: –89.5
    - **westernmost_longitude**: 0.0
    - **n_ cells**: 32400
- **horizontal_subgrid**
  - **cell_variable_type**: _x_velocity_
  - **horizontal_grid_cells**
    - **grid_type**: _regular_latitude_longitude_
    - **grid_mapping**: _latitude_longitude_
    - **region**: _global_
    - **temporal_refinement**: _static_
    - **x_resolution**: 1.0
    - **y_resolution**: 2.0
    - **horizontal_units**: _degree_
    - **southernmost_latitude**: –90.0
    - **westernmost_longitude**: 1.0
    - **n_ cells**: 32580

- **vertical_computational_grid**
  - **vertical_coordinate**: _atmosphere_hybrid_height_coordinate_
  - **n_z**: 85
  - **bottom_layer_thickness**: 10
  - **top_layer_thickness**: 810
  - **total_thickness**: 84763.34

6.4. _Grid: Tripolar ocean_

- **horizontal_computational_grid**
  - **arrangement**: _arakawa_c_
  - **horizontal_subgrid**
    - **cell_variable_type**: _mass_
    - **horizontal_grid_cells**
      - **description**: eORCA025
      - **grid_type**: _tripolar_
      - **grid_mapping**: _latitude_longitude_
      - **region**: _global_
      - **temporal_refinement**: _static_
      - **n_cells**: 1725200

- **vertical_computational_grid**
  - **vertical_coordinate**: *ocean_s_coordinate*
  - **n_z**: 75
  - **top_layer_thickness**: 1.5
  - **bottom_layer_thickness**: 200
  - **total_thickness**: 4000

6.5. *Grid: Reduced Gaussian*

- **horizontal_computational_grid**
  - **arrangement**: *arakawa_b*
  - **horizontal_subgrid**
    - **cell_variable_type**: *mass*
    - **horizontal_grid_cells**
      - **description**: T127. Gaussian Reduced with 256 grid points per latitude circle between 30 degrees north and 30 degrees south, reducing to 20 grid points per latitude circle at 88.9 degrees north and 88.9 degrees south.
      - **grid_type**: *reduced_gaussian*
      - **grid_mapping**: *latitude_longitude*
      - **region**: *global*
      - **temporal_refinement**: *static*
      - **y_resolution**: 1.40625
      - **horizontal_units**: *degree*
      - **southernmost_latitude**: –89.296875
      - **westernmost_longitude**: 0.9
      - **n_cells**: 24572
  - **horizontal_subgrid**
    - **cell_variable_type**: *x_velocity*, *y_velocity*
    - **horizontal_grid_cells**
      - **description**: T127. Gaussian Reduced with 256 grid points per latitude circle between 30 degrees north and 30 degrees south, reducing to 20 grid points per latitude circle at 88.9 degrees north and 88.9 degrees south.
      - **grid_type**: *reduced_gaussian*
      - **grid_mapping**: *latitude_longitude*
      - **region**: *global*
      - **temporal_refinement**: *static*
      - **y_resolution**: 1.40625
      - **horizontal_units**: *degree*
      - **southernmost_latitude**: –90
      - **westernmost_longitude**: 0.45
      - **n_cells**: 24572
- **vertical_computational_grid**
  - **vertical_coordinate**: *atmosphere_hybrid_sigma_pressure_coordinate*
  - **n_z**: 91
  - **bottom_layer_thickness**: 5
  - **top_layer_thickness**: 550
  - **total_thickness**: 61000

6.6. *Grid: Unstructured grid with two subgrids*

- **horizontal_computational_grid**

- **arrangement**: *arakawa_c*
- **horizontal_subgrid**
  - **cell_variable_type**: *mass*
  - **horizontal_grid_cells**
    - **description**: oEC60to30. Unstructured mesh created using Spherical Centroidal Voronoi Tessellations.
    - **grid_type**: *unstructured_polygon*
    - **grid_mapping**: *latitude_longitude*
    - **region**: *global*
    - **southernmost_latitude**: –90.0
    - **westernmost_longitude**: 0.435
    - **temporal_refinement**: *static*
    - **n_cells**: 235160
- **horizontal_subgrid**
  - **cell_variable_type**: *velocity*
  - **horizontal_grid_cells**
    - **description**: *oEC60to30.* Unstructured mesh created using Spherical Centroidal Voronoi Tessellations.
    - **grid_type**: *unstructured_polygon*
    - **grid_mapping**: *latitude_longitude*
    - **region**: *global*
    - **southernmost_latitude**: –89.9956
    - **westernmost_longitude**: 0.32
    - **temporal_refinement**: *static*
    - **n_cells**: 714274

6.7. *Grid: Cubed sphere*

- **horizontal_computational_grid**
  - *arakawa_c*
  - **horizontal_subgrid**
    - **cell_variable_type**: *mass*
    - **horizontal_grid_cells**
      - **description**: Grid cell edges vary in size between 100 and 111 km
      - **grid_type**: *cubed_sphere*
      - **grid_mapping**: *latitude_longitude*
      - **region**: *global*
      - **x_resolution**: 105
      - **y_resolution**: 105
      - **horizontal_units**: *km*
      - **southernmost_latitude**: –89.5
      - **westernmost_longitude**: 45
      - **temporal_refinement**: *static*
      - **n_cells**: 60000
  - **horizontal_subgrid**
    - **cell_variable_type**: *x_velocity*
    - **horizontal_grid_cells**
      - **description**: Grid cell edges vary in size between 100 and 111 km
      - **grid_type**: *cubed_sphere*
      - **grid_mapping**: *latitude_longitude*
      - **region**: *global*
      - **x_resolution**: 105
      - **y_resolution**: 105
      - **horizontal_units**: *km*

        - **southernmost_latitude**: –89.5
        - **westernmost_longitude**: 45
        - **temporal_refinement**: *static*
        - **n_cells**: 60000
- **horizontal_subgrid**
    - **cell_variable_type**: *y_velocity*
    - **horizontal_grid_cells**
        - **description**: Grid cell edges vary in size between 100 and 111 km
        - **grid_type**: *cubed_sphere*
        - **grid_mapping**: *latitude_longitude*
        - **region**: *global*
        - **x_resolution**: 105
        - **y_resolution**: 105
        - **horizontal_units**: *km*
        - **southernmost_latitude**: –89.8
        - **westernmost_longitude**: 43.1
        - **temporal_refinement**: *static*
        - **n_cells**: 60000

7. Controlled vocabularies

Many EMD property values are restricted to selections from controlled vocabularies, i.e. standardised lists which contain all possible values.

***Note**: It is known that some of these controlled vocabularies are not complete and will need expanding, because it is not known in advance what is required for every model. Subsequent versions of the EMD will include any new entries that are required.*

***Note for CMIP7**: The EMD on-line creation tool will provide the ability to request a new controlled vocabulary entry, which will be subsequently finalised on a GitHub issue. Once accepted, the new entry will be available for use within CMIP7, and the new term will appear in the next version of the EMD.*

*7.1. **component** CV*

Model component types.

A component type describes a set of interactive processes that are simulated by single model component. See sections 2 (Top-level model) and 3 (Model components).

Options for the top-level model **dynamic_components**, **prescribed_components**, and **omitted_components** properties; and the model component **component**, **embedded_in**, and **coupled_with** properties:

- **aerosol**
  - *The behaviour and evolution of aerosols suspended in the atmosphere.*
- **atmosphere**
  - *Dynamical, thermodynamical, and physical processes in the atmosphere.*
- **atmospheric_chemistry**
  - *The behaviour and evolution of the chemical composition of the atmosphere.*
- **land_surface**
  - *Water, energy, and mass fluxes between the surface and the atmosphere.*
- **land_ice**
  - *Frozen freshwater in glaciers, ice-caps, ice-sheets, and ice-shelves.*
- **ocean**
  - *Dynamical, thermodynamical, and physical processes in the ocean.*
- **ocean_biogeochemistry**
  - *Biological, geological, and chemical processes in the ocean.*
- **sea_ice**
  - *Frozen seawater that floats on the ocean surface.*

*7.2. **calendar** CV*

Calendar types.

A calendar defines a set of valid datetimes and their order that is used by the top-level model. See section 2 (Top-level model).

Options for the top-level model **calendar** property:

- **proleptic_gregorian**
  - *The calendar with the Gregorian rule for leap-years extended to dates before 1582-10-15, as defined by the CF conventions.*
- **standard**
  - *The Gregorian–Julian calendar, as defined by the CF conventions.*

- **julian**
  - *The calendar which follows the Julian rule for leap years, as defined by the CF conventions.*
- **360_day**
  - *The calendar in which all years are 360 days, and divided into 30 day months, as defined by the CF conventions.*
- **365_day**
  - *The calendar with no leap years, i.e. all years are 365 days long, as defined by the CF conventions.*
- **366_day**
  - *The calendar in which every year is a leap year, i.e. all years are 366 days long, as defined by the CF conventions.*
- **utc**
  - *The Gregorian calendar with leap seconds prescribed by Coordinated Universal Time (UTC), as defined by the CF conventions.*
- **tai**
  - *The Gregorian calendar without leap seconds that is based on International Atomic Time (TAI), as defined by the CF conventions.*
- **none**
  - *The designation for no calendar, as defined by the CF conventions.*

7.3. ***arrangement*** CV

Horizontal computational grid arrangement types.

A grid arrangement describes the relative locations of computed mass-related and velocity-related quantities on the horizontal computational grid (for instance Collins et al. (2013), and for unstructured grids Thuburn et al. (2009)). See section 4.1.1 (Horizontal computational grid).

Options for the horizontal computational grid **arrangement** property:

- **arakawa_a**
  - *The Arakawa A grid places mass- and velocity-related quantities at the same location on each grid cell.*
- **arakawa_b**
  - *The Arakawa B grid places velocity-related quantities at the corners of mass cells.*
- **arakawa_c**
  - *The Arakawa C grid places velocity-related quantities at the centres of mass cell edges, such that each component is perpendicular to its edge.*
- **arakawa_d**
  - *The Arakawa D grid places velocity-related quantities at the centres of mass cell edges, such that each component is tangential to its edge.*
- **arakawa_e**
  - *The Arakawa E grid places mass-related quantities at the centres of velocity cell edges.*

7.4. ***cell_variable_type*** CV

Types of physical variables that are carried at, or representative of conditions at, cells of a horizontal subgrid.

A variable type comprises a class of physical quantities that are typically found at the same grid location, i.e. at one of the locations defined by the horizontal computational grid **arrangement** property. See sections 4.1.1 (Horizontal computational grid) and 4.1.2 (Horizontal subgrid).

Options for the horizontal subgrid **cell_variable_type** property:

- **mass**
    - *Mass-related variables, including those representing thermodynamic and hydrodynamic quantities.*
- **x_velocity**
    - *Variables representing the X component of the velocity vector and certain horizontal fluxes of mass, energy, momentum, etc. For a grid defined by spherical polar coordinates, the X direction is longitude*.
- **y_velocity**
    - *Variables representing the Y component of the velocity vector and certain horizontal fluxes of mass, energy, momentum, etc. For a grid defined by spherical polar coordinates, the Y direction is latitude*.
- **velocity**
    - *Variables representing the velocity vector when it is not partitioned into X and Y components (such as for some unstructured grids)*.

7.5. *region* CV

Horizontal grid regions.

A region is a contiguous part of the Earth's surface which spans the horizontal grid cells, including areas for which no calculations are made, or data are available (such as ocean areas for a land surface component). See section 4.1.3 (Horizontal grid cells).

Options for the horizontal grid **region** property:

- **antarctica**
    - *The geographical region of Antarctica, as defined by the CF conventions*.
- **global**
    - *The geographical region of the whole of the Earth's surface, as defined by the CF conventions*.
- **greenland**
    - *The geographical region of Greenland, as defined by the CF conventions*.
- **limited_area**
    - *Any contiguous subregion of the Earth's surface, used to indicate a limited area model that may be placed over different geographical regions independently of the model formulation.*
- **30S-90S**
    - *The geographical region of the Earth's surface between 30 and 90 degrees south.*

7.6. *grid_type* CV

Horizontal grid types.

A grid type describes the method for distributing grid points over the sphere. See section 4.1.3 (Horizontal grid cells).

Options for the horizontal grid **grid_type** property:

- **regular_latitude_longitude**
    - *A rectilinear latitude-longitude grid with evenly spaced latitude points and evenly spaced longitude points.*
- **regular_gaussian**
    - *A Gaussian grid for which the number of longitudinal points is constant for each latitude.*
- **reduced_gaussian**
    - *A Gaussian grid for which the number of longitudinal points is reduced as the poles are approached.*

- **spectral_gaussian**
  - *A grid based on the transformation from spectral space to a reduced or non-reduced Gaussian grid.*
- **spectral_reduced_gaussian**
  - *A grid based on the transformation from spectral space to a reduced Gaussian grid.*
- **linear_spectral_gaussian**
  - *A spectral Gaussian grid for which the smallest spectral wavelength is represented by 2 grid points.*
- **quadratic_spectral_gaussian**
  - *A spectral Gaussian grid for which the smallest spectral wavelength is represented by 3 grid points*.
- **cubic_octahedral_spectral_reduced_gaussian**
  - *A spectral reduced Gaussian grid for which the smallest spectral wavelength is represented by 4 grid points, and which uses an octahedron-based method to reduce the number of grid points towards the poles.*
- **rotated_pole**
  - *A regular latitude-longitude grid that is rotated to define a different north pole location.*
- **stretched**
  - *A grid with higher resolution concentrated over an area of interest, at the expense of lower resolution elsewhere.*
- **displaced_pole**
  - *An ocean grid whose poles are not antipodean, typically with the northern pole displaced to lie over land.*
- **tripolar**
  - *A global curvilinear ocean grid with a southern pole and two northern poles all placed over land.*
- **cubed_sphere**
  - *The spherical surface is defined as six coupled “logically square” regions.*
- **icosahedral_geodesic**
  - *A grid that uses triangular tiles based on the subdivision of an icosahedron.*
- **icosahedral_geodesic_dual**
  - *A grid that uses hexagonal and pentagonal tiles and is the dual of an icosahedral_geodesic grid.*
- **yin_yang**
  - *Two overlapping grid patches.*
- **unstructured_triangular**
  - *An unstructured mesh consisting solely of triangles.*
- **unstructured_quadrilateral**
  - *An unstructured mesh consisting solely of quadrilaterals.*
- **unstructured_polygonal**
  - *An unstructured mesh consisting of arbitrary polygons.*
- **hierarchical_discrete_global_grid**
  - *A tessellation of cells that exhaustively cover the spherical surface of the globe, with each resolution defined as a subdivision of cells at a base resolution, and cells at a given resolution having equal areas*.
- **plane_projection**
  - *Any transformation employed to represent the spherical surface of the globe on a plane.*

7.7. ***grid_mapping*** CV

Horizontal grid mappings.

A grid mapping describes the horizontal coordinate reference system. See section 4.1.3 (Horizontal grid cells).

Options for the horizontal grid **grid_mapping** property:

- **albers_conical_equal_area**
  - *A projection that preserves area but not shape, creating accurate representations of large, east-west-oriented land masses, as defined by the [CF conventions](#).*
- **azimuthal_equidistant**
  - *A projection that maintains the correct distance and direction from a central point to all other points on the map, as defined by the [CF conventions](#).*
- **geostationary**
  - *A type of azimuthal projection designed to show the Earth as it appears to a geostationary satellite, offering a perspective that minimizes distortion for areas under view, as defined by the [CF conventions](#).*
- **healpix**
  - *The hierarchical equal area isolatitude pixelisation of a sphere based on subdivision of a distorted rhombic dodecahedron (HEALPix), as defined by the [CF conventions](#).*
- **lambert_azimuthal_equal_area**
  - *A projection that accurately preserves the area of all regions on a map, as defined by the [CF conventions](#).*
- **lambert_conformal_conic**
  - *A projection that preserves the shape of small areas by using a cone placed over the Earth's surface, with the projection onto the cone occurring along lines radiating from the Earth's centre, as defined by the [CF conventions](#).*
- **lambert_*cylindrical_equal_area***
  - *A cylindrical map projection that preserves area accurately but distorts shape, especially away from the equator, as defined by the [CF conventions](#).*
- **latitude_longitude**
  - *A projection that treats latitude and longitude as planar X and Y coordinates, as defined by the [CF conventions](#).*
- **orthographic**
  - *A form of parallel projection in which all the projection lines are orthogonal to the projection plane, as defined by the [CF conventions](#).*
- **polar_stereographic**
  - *An azimuthal projection that maps the Earth's surface onto a flat plane from a view point at one pole, projecting points from the globe to a plane tangent to the opposite pole, as defined by the [CF conventions](#).*
- **rotated_latitude_longitude**
  - *A geographic coordinate system where the traditional north pole is shifted to a different location, as defined by the [CF conventions](#).*
- **sinusoidal**
  - *An equal-area projection that accurately represents area but distorts shape, especially at the edges, as defined by the [CF conventions](#).*
- **stereographic**
  - *A projection that preserves the angles between intersecting lines and curves on the sphere, as defined by the [CF conventions](#).*
- **transverse_mercator**
  - *A projection that rotates the Mercator projection cylinder 90 degrees to be tangent to the Earth along a meridian instead of the equator, as defined by the [CF conventions](#).*
- **vertical_perspective**
  - *A projection used to represent the Earth from a point of view directly above its centre, as defined by the [CF conventions](#).*

*7.8. **temporal_refinement** CV*

Horizontal grid temporal refinement types.

A temporal refinement type describes how the distribution of grid cells varies with time. See section 4.1.3 (Horizontal grid cells).

Options for the horizontal grid **temporal_refinement** property:

- **static**
  - *The total number of grid points stays constant during the model run and there is no grid refinement, i.e. the grid is held fixed.*
- **dynamically_stretched**
  - *The total number of grid points stays constant, but grid points can be dynamically relocated.*
- **adaptive**
  - *The total number of grid points varies during the model run. The grid is refined locally when important physical processes occur that need additional grid resolution, and coarsened when the additional resolution is no longer needed.*

7.9. ***horizontal_units*** *CV*

Physical units of the horizontal grid **x_resolution** and **y_resolution** property values.

See section 4.1.3 (Horizontal grid cells).

Options for the horizontal grid **horizontal_units** property:

- **km**
  - *kilometre (unit for length).*
- **degree**
  - *degree (unit for angular measure)*.

7.10. ***truncation_method*** *CV*

Horizontal grid truncation method.

A truncation method describes the technique used to truncate the spherical harmonic representation of a spectral model. See section 4.1.3 (Horizontal grid cells).

Options for the horizontal grid **truncation_method** property:

- **triangular**
  - *Triangular truncation.*
- **rhomboidal**
  - *Rhomboidal truncation.*

7.11. ***vertical_coordinate*** *CV*

Vertical computational grid coordinate types.

A coordinate type describes the vertical coordinate reference system. See section 4.2.1 (Vertical computational grid).

Options for the vertical computational grid **vertical_coordinate** property:

- **air_potential_temperature**
  - *Air potential temperature is the temperature a parcel of air would have if moved dry adiabatically to a standard pressure, as defined by the CF conventions.*
- **air_pressure**

    - *Air pressure is the pressure that exists in the medium of air, as defined by the [CF conventions].*
- **atmosphere_hybrid_height_coordinate**
    - *Parametric atmosphere hybrid height coordinate, as defined by the [CF conventions].*
- **atmosphere_hybrid_sigma_pressure_coordinate**
    - *Parametric atmosphere hybrid sigma pressure coordinate, as defined by the [CF conventions].*
- **atmosphere_ln_pressure_coordinate**
    - *Parametric atmosphere natural log pressure coordinate, as defined by the [CF conventions].*
- **atmosphere_sigma_coordinate**
    - *Parametric atmosphere sigma coordinate, as defined by the [CF conventions].*
- **atmosphere_sleve_coordinate**
    - *Parametric atmosphere smooth vertical level coordinate, as defined by the [CF conventions].*
- **depth**
    - *Depth is the vertical distance below the earth's surface, as defined by the [CF conventions].*
- **geopotential_height**
    - *Geopotential height is the geopotential divided by the standard acceleration due to gravity, as defined by the [CF conventions].*
- **height**
    - *Height is the vertical distance above the earth's surface, as defined by the [CF conventions].*
- **land_ice_sigma_coordinate**
    - *Land ice (glaciers, ice-caps and ice-sheets resting on bedrock and also includes ice-shelves) sigma coordinate, as defined by the [CF conventions].*
- **ocean_double_sigma_coordinate**
    - *Parametric ocean double sigma coordinate, as defined by the [CF conventions].*
- **ocean_sigma_coordinate**
    - *Parametric ocean sigma coordinate, as defined by the [CF conventions].*
- **ocean_s_coordinate**
    - *Parametric ocean s-coordinate, as defined by the [CF conventions].*
- **ocean_s_coordinate_g1**
    - *Parametric ocean s-coordinate, generic form 1, as defined by the [CF conventions].*
- **ocean_s_coordinate_g2**
    - *Parametric ocean s-coordinate, generic form 2, as defined by the [CF conventions].*
- **ocean_sigma_z_coordinate**
    - *Parametric ocean sigma over z coordinate, as defined by the [CF conventions].*
- **sea_water_pressure**
    - *Sea water pressure is the pressure that exists in the medium of sea water, as defined by the [CF conventions].*
- **sea_water_potential_temperature**
    - *Sea water potential temperature is the temperature a parcel of sea water would have if moved adiabatically to sea level pressure, as defined by the [CF conventions].*
- **z***
    - *The z* coordinate, as defined by [Adcroft and Campin (2004)].*

8. EMD data model

The EMD may be represented by the [Unified Modelling Language] class diagram of **figure 2**.

All classes, apart from “Horizontal subgrid”, can exist independently of the others, as indicated by the hollow diamonds of their relationship connectors. The “Horizontal subgrid” class is different because it can only be understood in the context of its parent class “Horizontal computational grid”, as indicated by the filled diamond of the connector between those two classes.

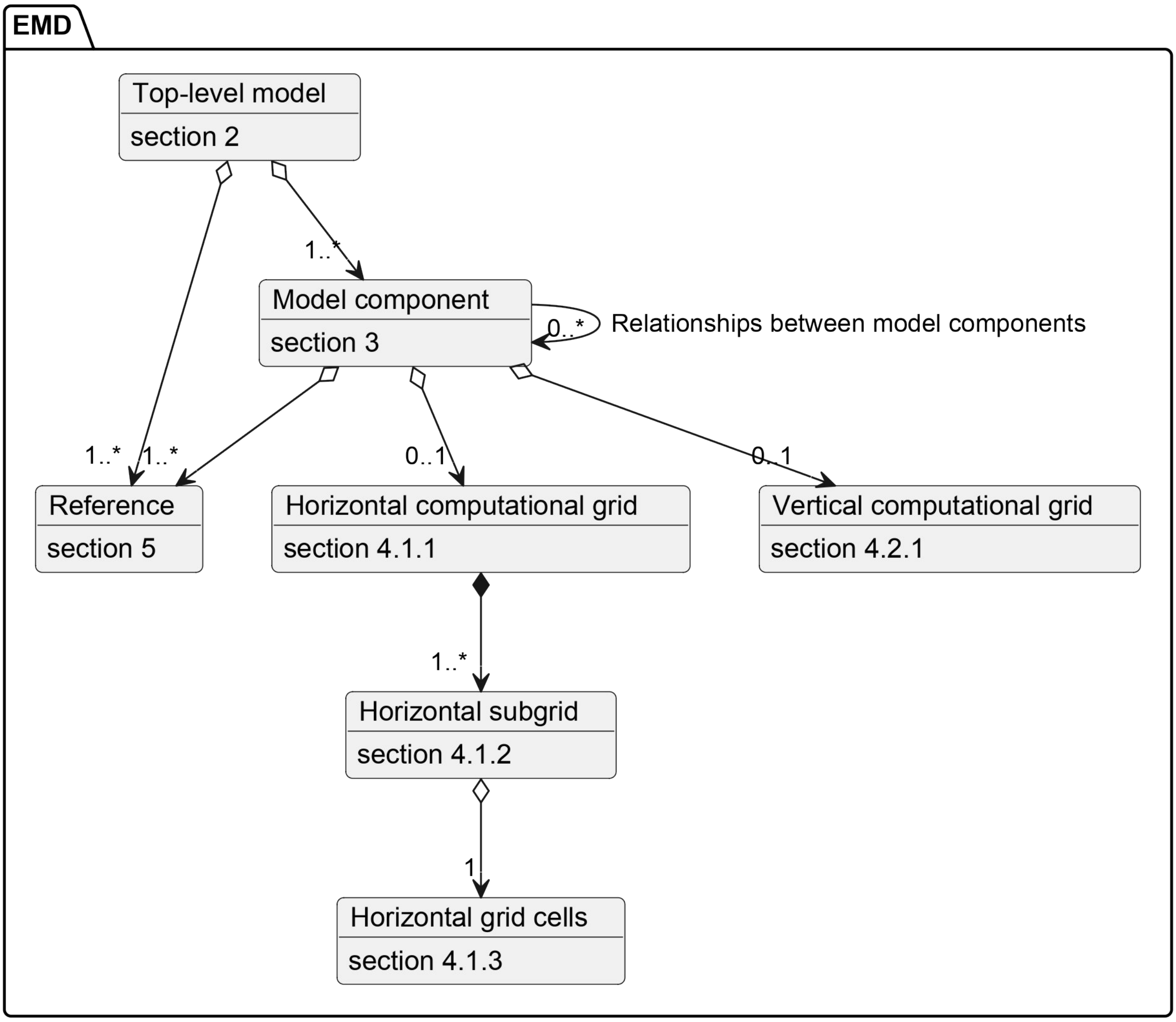

Figure 2 *The EMD data model. Each class indicates the section of this document where it is fully defined.*

Revision history

Version 1.0 (2025-12-03)

Improvements arising from the process of implementing EMD for CMIP7, made by the CMIP7 MIP Controlled Vocabularies Task Team.

- Providing horizontal and vertical grids to a model component is now optional.
- New horizontal and vertical grid structures.
- Clarification of coupled and embedded component definitions.
- New cell_variable_type controlled vocabulary.
- Updated the EMD data model.
- General text clarifications.

Version 0.992 (2025-05-16)

Hassell, D., Levavasseur, G., Pascoe, C., Taylor, K. E., Crawford, A., Durack, P. J., Ellis, D., Goddard, C., Horanyi, A., Lawrence, B., & Stockhause, M. (2025). Essential Model Documentation (EMD) (0.992). Zenodo. https://doi.org/10.5281/zenodo.15439552

The initial public release was developed by the CMIP7 Documentation Task Team from December 2023 to May 2025. During this period it was improved by suggestions arising from community reviews provided by:

- the WCRP Working Group on Coupled Modelling (WGCM) CMIP and WIP Panels,
- the CMIP7 MIP Controlled Vocabularies and Fresh Eyes on CMIP Task Teams,
- the CMIP Model Intercomparison Projects (MIPs),
- the climate modelling centres taking part in CMIP7,
- climate service providers acting as stakeholders for the archive of CMIP model outputs.

Bibliography

Adcroft, Alistair & Campin, Jean-Michel. (2004). **Rescaled height coordinates for accurate representation of free-surface flows in ocean circulation models**. Ocean Modelling. Volume 7, Issues 3–4, Pages 269–284. https://doi.org/10.1016/j.ocemod.2003.09.003

Arakawa, A., & Lamb, V.R. (1977). **Computational design of the basic dynamical processes of the UCLA general circulation model**. Methods in Computational Physics: Advances in Research and Applications. 17: 173–265. https://doi.org/10.1016/B978-0-12-460817-7.50009-4

Arakawa, A., & Moorthi, S. (1988): **Baroclinic instability in vertically discrete systems**. J. Atoms. Sci. 45 1688–1707. https://doi.org/10.1175/1520-0469(1988)045<1688:BIIVDS>2.0.CO;2

Balaji, V., & Liang, Zhi. (2007). **Gridspec: A standard for the description of grids used in Earth System models**. https://www.researchgate.net/publication/228641121

Berners-Lee, T., Fielding, R., & Masinter, L. (2005). **Uniform Resource Identifier (URI): Generic Syntax (RFC 3986)**. Internet Engineering Task Force. https://doi.org/10.17487/RFC3986

CF community (2025). **CF standard name table**. http://cfconventions.org/Data/cf-standard-names/current/build/cf-standard-name-table.html

CF community (2025). **CF standardized region list**. https://cfconventions.org/Data/standardized-region-list/standardized-region-list.current.html

Collins, S. N., James, R. S., Ray, P., Chen, K., Lassman, A., & Brownlee, J. (2013): **Grids in Numerical Weather and Climate Models**. In Climate Change and Regional/Local Responses. Edited by Zhang, Y. & Ray, P. InTechOpen. https://doi.org/10.5772/55922

Dunne, J. P., Hewitt, H. T., Arblaster, J., Bonou, F., Boucher, O., Cavazos, T., Durack, P. J., Hassler, B., Juckes, M., Miyakawa, T., Mizielinski, M., Naik, V., Nicholls, Z., O'Rourke, E., Pincus, R., Sanderson, B. M., Simpson, I. R., & Taylor, K. E. (2025): **An evolving Coupled Model Intercomparison Project phase 7 (CMIP7) and Fast Track in support of future climate assessment**, Geosci. Model Dev., 18, 6671–6700. https://doi.org/10.5194/gmd-18-6671-2025

Durack, P. J., Taylor, K. E., Mizielinski, M., Doutriaux, C., Nadeau, D., & Juckes, M. (2025). **CMIP6 Controlled Vocabularies (CVs) (6.2.58.76)**. Zenodo. https://doi.org/10.5281/zenodo.14657960

Eaton, B., Gregory, J., Drach, B., Taylor, K., Hankin, S., et al. (2025). **NetCDF Climate and Forecast (CF) Metadata Conventions (1.13)**. CF Community. https://doi.org/10.5281/zenodo.17801666

European Organization For Nuclear Research and OpenAIRE (2013). **Zenodo**. CERN. https://doi.org/10.25495/7gxk-rd71

Eyring, V., Bony, S., Meehl, G. A., Senior, C. A., Stevens, B., Stouffer, R. J., & Taylor, K. E. (2016). **Overview of the Coupled Model Intercomparison Project Phase 6 (CMIP6) experimental design and organization**, Geosci. Model Dev., 9, 1937–1958, https://doi.org/10.5194/gmd-9-1937-2016

Git Development Community (2025). **Git (Version 2.52.0)**. [Computer software]. https://git-scm.com

International DOI Foundation (2023). **DOI Handbook**. International DOI Foundation. https://doi.org/10.1000/182

Lawrence, B. N., Balaji, V., Bentley, P., Callaghan, S., DeLuca, C., Denvil, S., Devine, G., Elkington, M., Ford, R. W., Guilyardi, E., Lautenschlager, M., Morgan, M., Moine, M.-P., Murphy, S., Pascoe, C., Ramthun, H., Slavin, P., Steenman-Clark, L., Toussaint, F., Treshansky, A., & Valcke, S. (2012). **Describing Earth system simulations with the Metafor CIM**. Geosci. Model Dev., 5, 1493–1500, https://doi.org/10.5194/gmd-5-1493-2012

Masson, D., & R. Knutti (2011). **Climate model genealogy**. *Geophys. Res. Lett.*, 38, L08703, https://doi.org/10.1029/2011GL046864

Mesinger, F. (2001). **Chapter 13 numerical methods: The Arakawa approach, horizontal grid, global, and limited-area modeling**. International Geophysics. 70. 373–419. https://doi.org/10.1016/S0074-6142(00)80061-3

Moine, M.-P., Valcke, S., Lawrence, B. N., Pascoe, C., Ford, R. W., Alias, A., Balaji, V., Bentley, P., Devine, G., Callaghan, S. A., & Guilyardi, E. (2014). **Development and exploitation of a controlled vocabulary in support of climate modelling**. Geosci. Model Dev., 7, 479–493, https://doi.org/10.5194/gmd-7-479-2014

Object Management Group (OMG) (2017). **Unified Modeling Language (UML) Version 2.5.1**. OMG Document Number formal/2017-12-05, Dec. 2017. https://www.omg.org/spec/UML/2.5.1/PDF

Open Geospatial Consortium (2021). **Topic 21 - Discrete Global Grid Systems - Part 1 Core Reference system and Operations and Equal Area Earth Reference System**. OGC Abstract Specification 20-040r3. Edited by Robert Gibb. Wayland, MA: Open Geospatial Consortium. http://www.opengis.net/doc/AS/dggs/2.0

Taylor, K. E., Gleckler, P. J., Stouffer, R. J., Meehl, G. A., AchutaRao, K., Covey, C. C., Fiorino, M., Latif, M., McAvaney, B., Mitchell, J., Phillips, T. J., & Sperber, K. R. (2009). **CMIP3 Model Standard Output. In Program for Climate Model Diagnosis and Intercomparison**. Zenodo. https://doi.org/10.5281/zenodo.12765216

Taylor, K. E., & Doutriaux, C. (2010). **CMIP5 Model Output Metadata Requirements**. In Program for Climate Model Diagnosis and Intercomparison. Zenodo. https://doi.org/10.5281/zenodo.12764612

Taylor, K. E., Balaji, V., Hankin, S., Juckes, M., Lawrence, B., & Pascoe, S. (2012). **CMIP5 Data Reference Syntax (DRS) and Controlled Vocabularies (CVs). In Program for Climate Model Diagnosis and Intercomparison**. Zenodo. https://doi.org/10.5281/zenodo.12764404

Taylor, K. E., Juckes, M., Balaji, V., Cinquini, L., Denvil, S., Durack, P. J., Elkington, M., Guilyardi, E., Kharin, S., Lautenschlager, M., Lawrence, B., Nadeau, D., & Stockhause, M. (2018). **CMIP6 Model Output Metadata Requirements, Data Reference Syntax (DRS) and Controlled Vocabularies (CVs)**. In Program for Climate Model Diagnosis and Intercomparison. Zenodo. https://doi.org/10.5281/zenodo.12768887

Taylor, K. E., Troussellier, L., Ames, S., Hassell, D., Molina, M., Nicholls, Z., Schupfner, M., Anstey, J., Ellis, D., Dingley, B., Durack, P. J., Levavasseur, G., Mizielinski, M., & Moine, M.-P. (2025). **CMIP7 Global Attributes, DRS, Filenames, Directory Structure, and CVs (V1.0)**. Zenodo. https://doi.org/10.5281/zenodo.17250297

Taylor, K. E., & WGCM Infrastructure Panel (2025). **CMIP7 Guidance on Grids (v1.0)**. Zenodo. https://doi.org/10.5281/zenodo.15697025

Thuburn, J., Ringler, T. D., Skamarock, W. C, & Klemp, J. B. (2009). **Numerical representation of geostrophic modes on arbitrarily structured C-grids**. Journal of Computational Physics, Volume 228, Issue 22, Pages 8321–8335. https://doi.org/10.1016/j.jcp.2009.08.006